

## Patent Abstracts of Japan

PUBLICATION NUMBER : 62150477  
 PUBLICATION DATE : 04-07-87

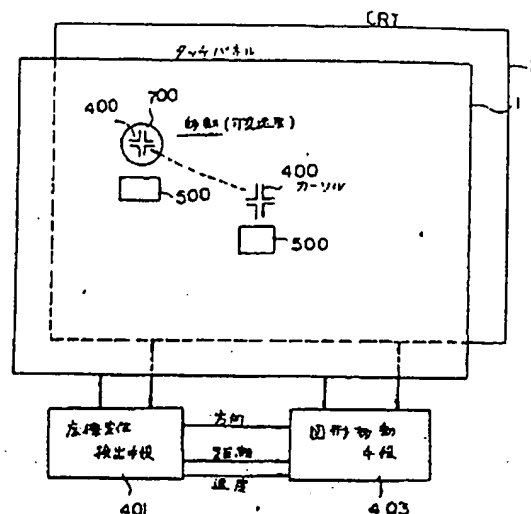
APPLICATION DATE : 25-12-85  
 APPLICATION NUMBER : 60290348

APPLICANT : CANON INC;

INVENTOR : YAMAKAWA TADASHI;

INT.CL. : G06F 15/62 G06F 3/14

TITLE : DISPLAY DEVICE



**ABSTRACT :** **PURPOSE:** To move a graphic form to a desired object accurately at a high speed by detecting inputted coordinate displacement on a touch panel and moving the graphic form to the same direction and distance in response to the detection speed.

**CONSTITUTION:** A transparent tablet 1 as a coordinate input means is overlapped on a display screen of a CRT display device 2. A coordinate display displacement detection means 401 and a graphic moving means 403 are connected to the CRT display device 2 and the transparent tablet 2. The movement of a finger inputted to a prescribed area on the touch panel 1, that is, the graphic moving command area 500 is detected by the coordinate displacement detection means 401 as the direction, distance and speed of the coordinate displacement and the graphic form 700 is moved in response to the detection speed.

**COPYRIGHT:** (C) JPO

## ⑫ 公開特許公報(A)

昭62-150477

⑪ Int. Cl.

G 06 F 15/62  
3/14

識別記号

庁内整理番号

6615-5B  
7341-5B

⑬ 公開 昭和62年(1987)7月4日

審査請求 未請求 発明の数 1 (全23頁)

⑭ 発明の名称 表示装置

⑮ 特 願 昭60-290348

⑯ 出 願 昭60(1985)12月25日

⑰ 発 明 者 山 川 正 東京都大田区下丸子3丁目30番2号 キヤノン株式会社内  
⑱ 出 願 人 キヤノン株式会社 東京都大田区下丸子3丁目30番2号  
⑲ 代 理 人 弁理士 大塚 康徳

## 明 細 書

## 1. 発明の名称

表示装置

## 2. 特許請求の範囲

(1) 略平面状の表示面をもち該表示面に、少なくとも移動対象の図形及び、該図形と別個の表示領域であつて前記図形を移動すべき指示を入力できる領域を表示する図形移動指示入力表示領域とを表示する表示手段と、前記表示面に重なるようにして配設された座標入力面をもつ座標入力手段であつて、前記図形移動指示入力表示領域に略一致した図形移動指示入力領域を前記座標入力面に有する座標入力手段と、前記図形移動指示入力領域に入力された入力座標の座標変位の方向、距離及び速度を検出する座標変位検出手段と、該座標変位検出手段により検出された前記変位速度に基

づいて変位距離の所定倍の距離だけ前記図形の表示位置を前記変位方向に平行移動する図形移動手段とを有する表示装置。

(2) 移動対象の図形は所定形状のカーソルである事を特徴とする特許請求の範囲第1項に記載の表示装置。

(3) 変位速度が所定の速度より大のときは前記座標変位の1倍以上の距離を、前記変位速度が所定の速度より小のときは前記座標変位の1倍未満の距離を対象の図形に移動させる事を特徴とする特許請求の範囲第1項又は第2項のいずれかに記載の表示装置。

(4) 図形と図形移動指示入力表示領域とは近接して設けられており、図形移動指示入力領域に入力された入力座標の座標変位と同一の移動を図形移動指示入力表示領域及び図形移動指示入力領域

に与えて前記図形と前記図形移動指示入力表示領域と前記図形移動指示入力領域とが少なくとも略同一方向に平行移動する事を特徴とする特許請求の範囲第1項乃至第3項のいずれかに記載の表示装置。

(5) 変位速度が小から大に急激に変化したときは、変化前後の速度の平均値から算出した移動距離を図形に与える事を特徴とする特許請求の範囲第1項に記載の表示装置。

ネルをCRT等の表示装置前面に重ねたもの等が考案されている。即ち、座標入力装置と表示装置とが一体となった座標入力／表示装置ともいうべきものである。このような装置において、画面上にメニューリストの各項目をいくつかの領域に分けて表示し、ペン先や指先で押された位置を含む領域の項目を選択するような場合には、当該領域が充分広いためにペン先や指先の入力座標値が目的とする位置より多少ずれても大きな問題はおこらない。

しかし、例えば昨今注目されているCAD (COMPUTER AIDED DESIGN)等の如き、図形作成等の精度が要求される分野においては種々の問題がある。第2図はCADによりプリント配線の設計を行う場合のCRT表示例である。CRT画面上には図示の如く横方向に密集したプリント配線が

### 3. 発明の詳細な説明

#### 【産業上の利用分野】

本発明は、例えばCRT等の表示手段に例えば透明デジタイザ等の座標入力手段を組合せた表示装置に関する。

#### 【従来の技術】

コンピュータによるデータ処理と画像処理技術とを組合せてグラフィックによるデータ入力分野では、オペレータが指示する位置に応じて表示変化をもたらすことにより、機械と人間とが会話的に作業を進められるマン＝マシンインターフェースの必要性が叫ばれている。

この種のマン＝マシンインターフェースには、透明デジタイザ又はタブレットの下に液晶表示器等の表示装置を重ねた組合せのもの、CRTとライトペンによる組合せのもの、又は透明タッチパ

走っている。さて、今カーソルが画面左上の点Aにあつた場合を想定してみる。点Aから配線密集領域内の点Bまでカーソルを移動する場合に、2つの方法がある。その第1の方法は軌跡Cの如くプリント配線の間を縫つてカーソルを移動しつつ点Bに到達する方法である。これは軌跡C上に新たなプリント配線を作図する事を想定すれば容易に了解できるであろう。そして、第2の方法は軌跡Dの如く略最短距離で点Bに到達する方法である。

#### 【発明が解決しようとする問題点】

上記2つの方法を分析して検討すれば、当該座標入力／表示装置には下記のような視点が必要なが分る。即ち、

①：カーソルを移動すべき方向、カーソルを移動すべき距離が把握できるような構成になつてい

こと、

②：座標入力観点から見れば目標位置を速くかつ正確に指示する事が課題である。これは現在のカーソルの位置から目標位置にカーソルを速やかに移動し、目標位置の近くにきたら正確にその位置を指示できなくてはならない。言い換えれば、目標点をより正確に指示できて、そのためには装置の操作者が目標点の位置を視認し易い構成になつてゐることである。又、目標への方角、目標までの距離を見失わない事も重大である。ところが、この観点に対して従来の座標入力指示具がスタイラスペンや指先などでは、その先端に表示されている図形がかくれてしまい、目的とする位置を的確に指示することが困難である。特にタッチパネル等のように座標入力装置の分解能がCRT等の表示手段の分解能に比較して小さい場合、C

が指示する点を軌跡として一次的に表示するという問題である。この点に関しても従来技術は、例えばタッチパネル等で入力できる座標の格子点の直下の点のみが正確に指示、即ち表示可能であるからして、不充分である。

③：更に正確に目標点に達するためには、又は所望の軌跡を描きながら目標点に達するためには、操作者の入力指示速度とカーソルの移動速度（又は、軌跡が描かれる表示描画速度）が、例えば目標点が遠いときは軌跡の描画速度を入力指示速度より速く、一方近くなつたときは前者の速度を後者の速度より遅くできる事が望ましい。

④：単なる表示装置又は座標入力装置としてではなく、以上の点を加味したマン・マシンインターフェースが望まれる事等である。

本発明は上記従来技術の問題点に鑑みてなされ

RT等に表示された任意の微小区画、例えば1ドットを正確に指示することはできない。又、タッチパネル等において、ある格子点とそのとなりにある格子点との間に表示されたドットを指示しようと思つても、場合によつては誤つた方の格子点の直下に表示された点を指示することになつてしまう。

⑤：①はある目標点に達するにはどうするかという観点からみた従来技術の問題点であつた。しかし第2図の軌跡Cの例でも分るように、ある目標点へ到達する事と同時にその目標点までの到達経路が問題になつてゐる。即ち、上記軌跡Cの例では既に表示されているプリント配線を避けつつ目標点Bに達するにはどうすればよいかということである。このカーソル位置を既存の画像から所定距離に保つ等という観点は、言い換えれば操作者

たものでその目的は、移動対象となる図形を所望の目標点に所望の移動速度で正確に移動可能な表示装置を提案するところにある。

#### 〔問題点を解決するための手段〕

上記課題を達成するための一手段として例えば第1図に示す実施例の表示装置は、表示手段であるCRTディスプレイ2の表示面と座標入力手段である透明タブレット1とを重ねるようになつてかつ両者の表示座標と入力座標とが所定の対応関係を有するよう重ねる。移動対象となる図形を図形700とした場合に、この図形700の特定は例えばカーソル400を図形700内に移動することで行う。表示装置には図形700を移動すべき指示を入力できる図形移動指示領域500がCRTディスプレイ2の表示面とタッチパネル1とで重なる領域をもつように設けられている。又、

表示装置には更に図形移動指示領域500に指等で入力された入力座標の座標変位の方向、距離及び速度を検出する座標変位検出手段401と、図形700を所望の位置に移動する図形移動手段403が接続されている。

#### 〔作用〕

上記構成において、タッチパネル1上の所定の領域、即ち図形移動指示領域500に入力されたときの指の移動を座標変位検出手段401が座標変位の方向、距離及び速度として検出し、その検出速度に応じて図形700を移動する、例えば変位速度が所定の速度より大のときは指の移動距離の1倍以上の距離を、変位速度が所定の速度より小のときは指の移動距離の1倍未満の距離を対象の図形700に移動させるようにする等するものである。このようにすると、図形700の移動は

操作者の移動指示領域500への例えば指による入力によりコントロールでき、更に図形700を移動すべき目標点に遠いときと近いときとで、図形700の移動速度を可変にできるので、操作性が良く、又所望の目標に高速かつ正確に移動できる。そして更に、この図形700の移動に追隨して、移動指示領域500もその位置を平行移動すると、更に操作性が向上する。

(以下余白)

#### 〔実施例〕

以下添付図面を参照しつつ本発明に係る実施例を更に詳細に説明する。

#### (実施例のブロック構成)

第3図(a)は実施例の表示機能付きの座標入力装置のブロック構成図である。1はタッチパネルでCRTディスプレイ2の画面上に画面を覆うように取りつける。CRTコントローラ3はCPU4のプログラムの実行により作成された画像データを画像メモリ5より取り出しCRTディスプレイ2上に表示する。CPU4のプログラム及びデータはメモリ6に格納する。また、タッチパネル1の面において指等で触れられた位置は、タッチパネル制御回路7によつて検出される。これはCPU4が座標値として読み出す。CPU4はこれらのデータをバス8を介してアクセスする。キ

ーボード9は画像表示、座標入力に必要な操作を入力するためのものである。例えば「モード」キー80、「入力」キー81等である。これらのキーの使用方法は後述する。

第3図(b)にCRT画面の例を示す。この画面の画素数は例えば横840ドット、縦400ドットとする。そして左上端の画素の座標を(0, 0)、右上端を(0, 633)、左下端を(399, 0)、右下端を(399, 633)とする。一般に、この様に座標づけした画素に対応するデータを画像メモリ5に格納する場合、横方向の8画素を1単位のメモリ位置、即ち1バイトに対応させ、横方向にシーケンシャルに格納し、1ドットライン分の次の1ドットラインをシーケンシャルに格納していき、1画面分のデータをラストイメージで格納する方法が一般にとられる。この様

にして、CPU 4から画像メモリ5の内容を通常のメモリ6と同様に取り扱えるようにする。以上の様な環境のもとで、CRTディスプレイ2の画面上の任意位置に点を形成する。任意位置から任意位置に線を引く。任意位置を中心に円をえがく、等の描画プログラムの他、画面上の任意矩形領域のビットパターンデータを特定の形式でメモリ6中に転送したり、その逆でメモリ6中のデータを画面の任意の位置に出力する等の描画プログラムがある。これはCPU 4が画像メモリ5の内容を書き換えたり、メモリ6との間でデータ転送することにより実現している。この画像メモリ5の内容を書き換える際にそれぞれの画素データに対応するビット単位で、書き換えのソースデータをS、画像メモリの画素データをDとしたとき、 $D = \alpha \cdot S$ なる論理演算 $\alpha$ を施した結果を画像メモ

スプレイ2上の座標と、画像メモリ5内の座標とがユニークに対応しているので、CRT画面をみながら画素上を押したときのタッチパネル1上の座標値を入力し、その値から、画像メモリ5の座標(x, y)を算出できればよい。

#### (グラフィックカーソル)

さて第4図は実施例に用いられるグラフィックカーソルの原理的構成例を示す。CRTディスプレイ2の表示面にはし型の図形を4つ合せたカーソル10とカーソル移動指示表示領域11と座標入力指示表示領域12という3つの表示領域が設けられている。又、タッチパネル1の座標入力面には前記カーソル移動指示表示領域11に対応したカーソル移動指示入力領域14と、座標入力指示表示領域12に対応した座標入力指示入力領域13の2つの領域が設けられている。操作者はカ

りの新データとする手法などともとられる。例えば、画面の反転であれば、 $\alpha$ を排他的論理和、Sを"1"とすることにより実現できる。

以上はCPU 4のプログラムとして図形処理を行なう例であるが、CRTコントローラ3にこれらの機能を有するものもあり、この場合、CPU 4は描画指令をCRTコントローラ3に出すだけでよい。

一方画面上の任意の点を指等で押すと、画面上にはタッチパネル1が設置されているので、その座標位置をCPU 4はタッチパネル制御回路7から読み出すことができる。タッチパネル制御回路7は押された位置のX方向の座標とY方向の座標とを数値コードでCPU 4に受け渡してもよいし、2進数値として受け渡してもよい。いずれにおいても、タッチパネル1の座標と、CRTディ

スプレイ2の中心部Pを所望の表示位置に移動するために、カーソル移動指示表示領域11を視認しながらカーソル移動指示入力領域14内で移動方向及び移動距離を、例えば指又はスタイラスペン等で与える。又、座標入力指示表示領域12の表示を視認しながら座標入力指示入力領域13を押すことにより、カーソル10の中心部Pの示す画素を新たな画素(ドット)として入力すべき事を指示する。

このように構成されたカーソルを本実施例では「グラフィックカーソル」と呼ぶこととする。第4図に示したように5つの構成要素からなるグラフィックカーソルを適当に組合せて色々な操作が可能となる。例えば

①：カーソル10と、カーソル移動指示表示領域11とカーソル移動指示入力領域14とを、カー

ソル10がカーソル移動指示入力領域14を指で押しても隠されない程度に近接した領域に設定するようにする。カーソル移動指示表示領域11上にて指を動かしてカーソル10を移動させると、カーソル移動指示表示領域11とカーソル移動指示入力領域14もカーソル10と一緒に移動するようにする。この操作は指の移動につれてカーソル10が指と短い距離を保つたまま移動するものであり、しかもカーソル10は指から離れているので指で隠されることはないから、極めて操作性が高い。目標点まで正確かつ要領良く到達できるので表示装置としての価値が高い。

②：①の構成に加えて座標入力指示表示領域12と座標入力指示入力領域13もカーソル10に近接させて設定し、かつカーソル10が移動するときはこの領域も移動する。この手法では①に述べ

た特徴に加えて、座標入力という点で極めて操作性が良くなる(この例として、第5図(a)、(b)に示す)。

③：以上2つの構成はカーソル移動指示表示領域11及び座標入力指示表示領域12がグラフィック表示されるものであつた。それらに対して、例えば座標入力指示表示領域12と座標入力指示入力領域13をキーボード9上の例えば前述の「入力」キー81等で代用させてもよい。又、

④：カーソル移動指示表示領域11とカーソル移動指示入力領域14とをカーソル10と一緒に移動させないで、第6図(c)の如く固定領域に設定してもよい。

⑤：又、カーソルの代りに図形一般を移動対象とすることもできる。

(グラフィックカーソルの具体例)

グラフィックカーソルの構成の2つの例を第5図(a)、(b)に示す。

第5図(a)のグラフィックカーソル100では、カーソル10と座標入力指示表示領域12、座標入力指示入力領域13とを一致させて重ね(このように重ねた領域を座標入力指示領域30と呼ぶことにする)、座標入力指示領域30の近くにカーソル移動指示表示領域11とカーソル移動指示入力領域14とを重ねた移動指示領域40を設定する。第5図(a)のグラフィックカーソル100の全体は水平長が $L_{1x}$ 、垂直長が $L_{1y}$ の矩形領域に形成される。L字形の領域20a、20b、20c、20dはカーソル10を構成し、特にこのような構成にしたのもその中央部Pを特定し易くしたためである。従来のグラフィックカーソルと本実施例のグラフィックカーソルとの差

異を隔立たせるために、この4辺のL字形の中心を以後「代替位置」と呼ぶ。カーソル10と移動指示領域40とを分離して、指等による入力に対してカーソルの中心部Pの座標が「代りに」入力座標となる本実施例の特徴からである。

グラフィックカーソル100の基準座標位置を左上隅( $X_0, Y_0$ )とすると、代替位置Pの相対的長さは水平距離が $\Delta X_1$ 、垂直距離が $\Delta Y_1$ である。即ち、代替位置Pは( $X_0, Y_0$ )を原点として、( $\Delta X_1, \Delta Y_1$ )で表わされる。移動指示領域40は四角形で表わされ、その領域の大きさは水平長が $L_{4x}$ 、垂直長が $L_{4y}$ 、又、当該領域の基準点である左上隅位置は( $X_0, Y_0 + L_{4y}$ )となるようにしてある。また、座標入力指示領域30の基準位置は( $x_0, y_0$ )にとり、その水平長は $L_{3x}$ 、垂直長 $L_{3y}$ とする。このよう

に各図形の基準座標とそこからの距離を与えれば、所望のグラフィックカーソル図形を描く事は容易である。即ち、基準位置を移動させていけばグラフィックカーソル100はそのまま移動することになる。

第5図(b)は第5図(a)のグラフィックカーソル100とは若干異なつた機能を有するグラフィックカーソル200の例である。このグラフィックカーソル200は水平長が $L_x$ 、垂直長が $L_y$ の矩形領域に形成される。代替位置Qのオフセットは水平が $\Delta X$ 、垂直が $\Delta Y$ である。グラフィックカーソル100と特に異なるのは、移動指示領域60がグラフィックカーソル200全体に設定しているものであり、従つて座標入力指示領域50(この領域は同時に移動指示領域でもある)は移動指示領域60から移動のみを指示す

100又は200全体が移動する。この移動は基準座標( $x_0, y_0$ )の値を変える事によりなされる。図を見てもわかるように、移動中は指で代替位置Pは指で隠されないことに留意すべきである。代替位置Pが目標位置に到達すれば座標入力指示領域30を指で押すことにより、その目標位置の座標が入力される。

第6図(b)は「モード」キー80の操作により、連続入力モード(第11図の制御手順)として本装置を使用するときの図である。この連続入力モードではグラフィックカーソル100の移動指示領域40に指を当てながら移動すると、グラフィックカーソル100は指の移動につれて移動するのはもちろんであるが、同時に代替位置Pの座標が入力されて連続した座標入力(それをそのまま表示すれば軌跡になる)が得られる。

る領域を表わす四角形70を除いた、グラフィックカーソル200の残りの領域である。尚、この座標入力指示領域50の水平長は $L_{4x}$ 、垂直長は $L_{4y}$ で、基準位置からのオフセットは水平、垂直とも“0”である。このグラフィックカーソル200はグラフィックカーソル100の機能に加えて、座標を連続して入力することが可能であるように、座標入力指示領域50を代替位置Qが隠れないようにして押しつつ移動させると移動指示と座標入力指示が同時に可能になるものである。

#### (グラフィックカーソルの操作)

第6図(a)にグラフィックカーソル100(第5図(a))の操作方法の一例を示す。CRTディスプレイ2上でのカーソル出発位置で、移動指示領域40を押しながら、指を離さずに指を移動する。この移動につれてグラフィックカーソル

第6図(c)の例は第4図の各構成要素がばらばらである構成であつて、カーソル10が従来のようにCRTディスプレイ2上を移動するのであるが、カーソル移動指示表示領域11、座標入力指示表示領域12、座標入力指示入力領域13、カーソル移動指示入力領域14とが移動せずに、図示の固定的な位置に留まるものである。

第6図(d)の例は座標入力指示をキーボード9上の座標入力キー81により行うものである。

#### (その他のグラフィックカーソルの例)

第5図(c)、(d)に他のグラフィックカーソル300、600の例を示す。第5図(c)のように外枠301と内枠302と間の領域を移動のみの指示領域とし、内枠302の内部が入力指示領域304、外枠301の内部が移動指示領域303である。尚、代替座標Rは前述と同様に4



つのし字形の中心で表わされる。このように代替座標のまわりを入力指示領域とし、そのまわりを移動指示領域とすることにより、代替座標Rを画面の上下左右端の任意の位置に移動できるようにする。第5図(d)のように、代替座標Sのまわり以外のところに、入力指示領域801を定義してもよい。特に、グラフィックカーソルの大きさが限られる場合、代替座標Sから離れた場所でも、軌跡入力できるようにする場合に有効になる。もちろん、領域802を移動のみとして、領域801とし字形付近を入力指示領域とするようにしてもよい。

#### (描画プログラムと座標入力プログラム)

第7図(c)はCRTディスプレイ2上に回路図を描画する一例を表わしている。点Eからゲート101の入力ピンFまで直線(プリント配線)

6内での配置は第7図(b)のようである。第7図(c)は描画プログラムと座標入力プログラムとのインターフェースを示す図である。座標入力プログラムからは、ONOFF変化フラグ、OFFON変化フラグ、スタート座標、エンド座標、ON座標、ONフラグ等が描画プログラムに渡される。これらの情報は座標入力プログラムがメモリ6内に格納する(第8図参照)。描画プログラムと座標入力プログラムとは多重処理される。描画プログラムはメモリ6に格納されたこれらの情報を読み出して、自分のプログラムの制御手順に沿って描画していく。

#### (制御情報)

第8図は実施例の制御に必要なフラグ等がメモリ6に格納されている様子を示している。順に、仮座標90、移動中フラグ91、基準位置座標9

を引くものである。点Fはゲート102の入力ピンとは近接しており従来のグラフィックカーソルでは点Fと点Gとを区別してカーソルを移動させる事は困難であつたのは前述した通りである。この描画プログラムの一例は第9図の如くである。さて描画プログラムに描画させるには、入力座標等が必要であるが、そのような情報を与えるのが座標入力プログラムである。座標入力プログラムはカーソルを移動したり、入力座標を描画プログラムに与える役目をもつ。座標入力プログラムの例を第10図、第11図、第12図、第13図、第14図、第15図に示す。座標入力プログラムはCPU4により所定の時間間隔のタイマ値105(第8図)毎に呼び出されて、描画プログラムとは独立して実行される。

#### 描画プログラムと座標入力プログラムのメモリ

2、旧座標93、ONフラグ94、代替位置座標95、移動領域内フラグ96、入力中フラグ97、連続モードフラグ98、OFFON変化フラグ99、ONOFF変化フラグ101、スタート座標102、エンド座標103、ON座標104、タイマ値105等であり、これらの具体的な説明は制御手順説明時に行う。又、前述のグラフィックカーソルの大きさ等を特定するL<sub>1</sub>等の量もメモリ6に格納される。

#### (描画プログラムの制御手順)

第9図は描画プログラムの制御手順である。第7図(a)に示したような直線を引くための簡単なプログラムである。ステップS2ではこの描画プログラムに必要なメニューを表示する。ステップS4ではOFFON変化フラグ99がセットされるのを待つ。このOFFON変化フラグ99は

座標入力プログラムでセットされるもので、このフラグが“1”であることは指等でカーソル移動指示入力領域14が初めて押された事を示す。この時、後述する座標入力プログラムによりメモリ6内にスタート座標102が格納されている。ステップS6ではONOFF変化フラグ101がセットされるのを待つ。このONOFF変化フラグ101はカーソル移動指示入力領域14から指が離されたことを示すフラグである。このフラグが“1”のときにはメモリ8にはエンド座標103が格納されている。ステップS8ではスタート座標102とエンド座標103に対応する画像メモリ5内のアドレスを算出し、その間にある画素を全て“1”とする。こうして第7図(a)に示された点Eと点Fは簡単に直線で結ばれたことになる。

ラグ94は指がタッチパネル1のいずれかの点を前回に押していた事を示す。ONフラグ94が“0”のときは座標入力プログラムを終了し、“1”の場合はステップS98で、ONOFF変化フラグ101を“1”にする。ステップS100で、エンド座標103としてON座標104を、即ち前回の入力座標値で、指が離れる直前の座標値を登録する。そしてステップS102でONフラグ94を“0”にして座標入力プログラムを終了する。所定時間経過後には本座標入力プログラムは再び実行される。

ステップS80でタッチパネル1のいずれかの点を押されている場合に、入力有りとしてステップS62に進み、そのタッチ座標106(X', y')を入力する。このとき、タッチパネル1の制御方式によつては、まず座標値(X', y')

(座標入力プログラム)

次に本実施例に最も特徴的な座標入力プログラムの例を以下に説明する。前述したように、この座標入力プログラムは定期的に所定時間間隔(タイマー値105)毎に呼び出され実行される。

(グラフィックカーソル100の制御手順)

第10図は第5図(a)に示したグラフィックカーソル100の操作に伴う制御手順である。ステップS60でタッチパネル1のいずれかの場所を押されたかどうか判断する。ステップS60で入力なしと判断したときはステップS94に進む。指が触れていないので、ステップS94で移動中フラグ91を“0”にする。この移動中フラグ91はカーソル移動指示入力領域14が一度は押された事を示すものである。ステップS96ではONフラグ94をチェックする。このONフ

を入力し、この値の範囲チェックにより入力の有無を判断してもよい。ステップS64では、タッチパネル1と画像メモリ5との対応に従った座標変換をほどこし、画面上の座標に変換し仮座標90として登録する。ステップS66では前回のサンプリング時に移動指示領域内に指が触れられたかどうかを示す移動中フラグ91をチェックし、このフラグが“0”のとき、即ち、前回のサンプリング時にカーソル移動が指示されていないときはステップS72に進む。ステップS72では仮座標90(これを(X, y)とする)と、(x, y)が移動指示領域40内かどうかを判断する。この判断は、基準位置を(x<sub>0</sub>, y<sub>0</sub>)とする

$$\begin{cases} x_0 \leq x \leq x_0 + l_{xx} - 1 \\ y_0 + l_{yy} \leq y \leq y_0 + l_{yy} + l_{yx} - 1 \end{cases}$$

のとき移動指示領域40内と判断する。

この領域外のときステップS76に進み、移動中フラグ91を“0”にしてステップS80で、仮座標90が座標入力指示領域30内かどうかを判断する。このチェックは、

$$\begin{cases} x_b \leq x \leq x_b + \Delta x_{1y-1} \\ y_b \leq y \leq y_b + \Delta y_{1y-1} \end{cases}$$

のとき座標入力指示領域30内と判断する。座標入力指示領域30内のときはステップS84にてON座標104を代替座標95、即ちカーソル内の点Pの座標で登録する。描画プログラムに渡されるべき情報であるON座標104を(x', y')とすると、

$$\begin{cases} x' = x_b + \Delta x_1 \\ y' = y_b + \Delta y_1 \end{cases}$$

と表わされることになる。ステップS80にて仮座標90(タッチされた点)が座標入力指示領域

30外のときは、ステップS82にてON座標104を仮座標90の値で登録する。これは操作者がカーソルを使う以外の目的でタッチパネル1にタッチしたものと考えたからである。

そしてステップS86に進む。ここではONフラグ94をチェックする。ONフラグ94は前回のサンプリングで座標入力されたことを表わすフラグであり、これが“1”ならばプログラムを終了し、“0”ならば前回サンプリング時オフ、今回オンであるので、その変化を処理するためにステップS88に進む。ここでON状態への変化が発生したことを表わすOFFON変化フラグ99を“1”にし、ステップS90でスタート座標102をON座標104の値で登録し、ステップS92でONフラグ94を“1”にしてプログラムを終了する。

ステップS72で移動指示領域40内と判断したときは、ステップS74で移動中フラグ91を“1”にして、ステップS98に進む。ここで仮座標90の値を旧座標93として登録する。これは後述するステップにて、移動指示領域40に指を触れたまま指を動かすことによりグラフィックカーソル100を移動させるために、もとの指の位置を記憶しておくためのものである。

ステップS66にて移動中フラグ91が“1”のときは、前回移動指示領域40に触れており、現在も画面に触れていることになる。従つて、ステップS68、S70に進み、グラフィックカーソル100を指の移動に従つて移動させる。即ち、ステップS68ではグラフィックカーソル100の基準座標の新しい値を算出する。これは指の移動変位をそれぞれx座標、y座標で $\delta x$ 、 $\delta y$

とすると、

$$\begin{aligned} \delta x &= (\text{仮座標90のx座標値}) - (\text{旧座標93のx座標値}) \\ \delta y &= (\text{仮座標90のy座標値}) - (\text{旧座標93のy座標値}) \end{aligned}$$

であるから、グラフィックカーソル100の新たな基準座標( $x_b$ ,  $y_b$ )を

$$\begin{cases} x_b = x_b + \delta x \\ y_b = y_b + \delta y \end{cases}$$

と更新する。そしてステップS70でグラフィックカーソル100の移動処理を行なう。即ち、グラフィックカーソル100の表示機構(不図示のグラフィックカーソル表示回路)に対し、新しい基準座標の更新を行なえばよい。次に、ステップ

S 9 8で次の基準位置移動に備えて、この仮座標 9 0の値を旧座標 9 3に移す。

以上のように構成すると、任意の位置の座標は移動指示領域 4 0に触れてそのまま指を動かして、4つのL字形領域の中心、即ち代替位置 Pをその所望の位置に設定したのち座標入力指示領域 3 0を指で触れることにより可能になる。

(グラフィックカーソル 2 0 0の制御手順)

以上説明したグラフィックカーソル 1 0 0の構成では、正確な軌跡を入力することができない。これに対応するには、第5図(b)の如くグラフィックカーソル 2 0 0全体を移動指示領域とみなし、その中の一部を確定指示領域 5 0とする。このようなグラフィックカーソル 2 0 0の操作に係る制御手順を第11図に示す。

ステップ S 1 1 0で、タッチパネル 1が押され

いないとき、ステップ S 1 2 2に進む。ステップ S 1 2 2では、仮座標 9 0 (x, y)がグラフィックカーソル 2 0 0内の座標かどうかを判断する。この判断は第5図(b)を参照しつつ、

$$\begin{cases} x_0 \leq x \leq x_0 + L_x - 1 \\ y_0 \leq y \leq y_0 + L_y - 1 \end{cases}$$

のときグラフィックカーソル 2 0 0内であると判断すればよい。

今、仮座標 9 0がグラフィックカーソル 2 0 0内にない場合はステップ S 1 3 0に進み、移動領域内フラグ 9 6を"0"にして、ステップ S 1 3 2で座標入力中フラグ 1 0 7を"0"にする。座標入力中フラグ 1 0 7はグラフィックカーソル 2 0 0の座標入力指示領域 5 0に指が触れられていた事を表わすものであるから、グラフィックカーソル 2 0 0に触れていないため"0"クリアす

たかどうか判断する。押されていない場合の制御手順は第10図のステップ S 9 4～ステップ S 1 0 2と略同じであるので説明は省略する。

タッチパネル 1が押されていると判断した場合入力有りとしてステップ S 1 1 2に進み、そのタッチ座標 1 0 6 (x', y')を入力する。ステップ S 1 1 4ではタッチ座標 1 0 6 (X', Y')に前記実施例のステップ S 6 4で行ったのと同じ座標変換を施し画面上の座標に変換し、仮座標 9 0として登録する。ステップ S 1 1 6では、前回のサンプリング時にグラフィックカーソル 2 0 0内の何処にか指が触れられていたのかどうかを示す移動領域内フラグ 9 6をチェックする。

移動領域内フラグ 9 6が"1"のとき、即ち、前回はグラフィックカーソル 2 0 0上を指示して

る。そこで現在、指はグラフィックカーソル 2 0 0以外の場所を触れていることになるので、この位置を入力座標値とするため、ステップ S 1 4 6でON座標 1 0 4の値を仮座標 9 0の値で登録する。そしてステップ S 1 5 4ではONフラグ 9 4をチェックする。ONフラグ 9 4が"1"ならばプログラムを終了し、"0"ならば前回のサンプリング時にオフ、今回はオンであるので、ステップ S 1 5 8に進む。ここでタッチパネル 1へのON状態への変化が発生したことを表わすOFFON変化フラグ 9 8を"1"にし、ステップ S 1 5 8でスタート座標 1 0 2をON座標 1 0 4の値で登録し、ステップ S 1 6 0でONフラグ 9 4を"1"にしてプログラムを終了する。

ステップ S 1 2 2にてグラフィックカーソル 2 0 0内を指がさしているとは判断したときは、ステ

ステップS124に進み、まず移動領域内フラグ96を“1”にする。さらにステップS126にて仮座標90が座標入力指示領域50内かどうかをチェックし、座標入力指示領域50内になっているときのみ、ステップS128で座標入力中フラグ107を“1”にする。そうでないときはこの107はステップS132、S136でクリアされているため“0”のままである。ステップS128の座標入力指示領域50内かどうかのチェックは、

$$\begin{cases} x_0 \leq x \leq x_0 + L_x - 1 \\ y_0 \leq y \leq y_0 + L_y - 1 \end{cases}$$

の式に基づいて判断すればよい。

そしてステップS148に進み、ここで仮座標90の値を旧座標93として登録する。このステップS148の意味は前述のステップS78と同

ことができ、指を離したときのOFF状態への変化も同様に入力可能となる。

ステップS116にて移動領域内フラグ96が“1”のときは、前回グラフィックカーソル200に触れており、現在も画面に触れていることになる。従ってステップS118、S120と進みグラフィックカーソル200を指の移動に従って移動させる。ステップS118ではグラフィックカーソルの基準座標の新しい値を算出する。これは、指の移動変位をそれぞれX座標、Y座標で $\delta x$ 、 $\delta y$ とすると、

$$\begin{aligned} \delta x &= (\text{仮座標90の}x\text{座標値}) - (\text{旧座標93の}x\text{座標値}) \\ \delta y &= (\text{仮座標90の}y\text{座標値}) - (\text{旧座標93の}y\text{座標値}) \end{aligned}$$

$$\begin{aligned} \delta x &= (\text{仮座標90の}x\text{座標値}) - (\text{旧座標93の}x\text{座標値}) \\ \delta y &= (\text{仮座標90の}y\text{座標値}) - (\text{旧座標93の}y\text{座標値}) \end{aligned}$$

じである。ステップS150では座標入力中フラグ107が“1”かどうかを調べる。このフラグが“0”であるときは、即ち座標入力指示領域50に触れていない場合は、本プログラムを終了し、触れている場合はステップS152にてグラフィックカーソル200の代替位置の座標値をON座標104の値として登録する。この座標指示位置の座標( $x'$ ,  $y'$ )は以下のようにして算出できる。

$$\begin{cases} x' = x_0 + \Delta X_0 \\ y' = y_0 + \Delta Y_0 \end{cases}$$

これにより指で押した位置ではなく、グラフィックカーソル200で示した点Qの位置の座標をON座標104として用いることが可能となる。そしてステップS154以下を前述のように実行すれば、ON状態への変化も同様にしてとらえる

座標値)

であるから、グラフィックカーソル200の基準座標( $x_0$ ,  $y_0$ )を

$$\begin{cases} x_0 = x_0 + \delta x \\ y_0 = y_0 + \delta y \end{cases}$$

とすればよい。そしてステップS120で、グラフィックカーソル200の移動処理を行なう。

そしてステップS148以下を前述のように実行することにより、座標座標入力指示領域50に触れたまま指を移動させて、グラフィックカーソル200内の点Qの座標指示位置の座標を連続して入力することも可能となる。即ち、描画プログラム(第9図とは異なる)として、座標入力プログラムから得たON座標104を連続的に画像メモリ5内に展開するように組めば、座標入力軌跡として得られることになる。前述の第7図

(a) を実行する座標入力プログラム (第10図) は特に表示装置としてみた場合に効果的であるが、第11図の座標入力プログラムは座標入力装置のプログラムとしての意義がある。

#### (連続座標入力)

第12図は第5図(a)のグラフィックカーソル100を使つて、連続的に座標入力する変形例である。その為にキーボード9上の「モード」キー80を操作して連続入力モードに切替える(即ち、連続モードフラグ98が「1」となる)。第10図のフローチャートとの相違はステップS79のみで、このステップS79で連続モードフラグ98を調べ、このフラグが「1」であればステップS84へ進むが、「0」であるときはプログラムを終了させるというものである。

更に、「モード」キー80の代りに画面上にモ

ード切替指示のための領域を設定することもできる。その場合、例えば移動指示領域40、60内の一部にモード切替指示領域を設定し、この領域へのOFF→ON変化をチェックするステップを、ステップS66とS72との間に入れ、その領域でのOFF→ON又はON→OFF変化の度に連続モードフラグ98を反転させるようにする。一方、移動指示領域40、60外にモード切替指示領域を設定する場合は、OFF→ON又はON→OFF変化チェックするステップをステップS80とステップS82との間に入れる。但しこの場合、モード切替指示があつた場合、ステップS82には進まずプログラムを終了するようにすればよい。

こうして、座標入力の方式を単発か連続かのいずれかに適宜変更しながら、かつ代替位置座標を

入力座標として座標入力可能となるので、操作性が大幅に向上する。

#### (キーボードからの座標入力指示)

以上説明した実施例は座標入力指示を表示図形上への入力(即ち、タッチパネル1への入力)により行うものであつた。第13図に示す実施例は第6図(d)に一部対応するもので、座標入力指示をキーボード9から行うものである。この指示は第6図(d)にも示す如く「入力」キー81によりなされるものである。第13図に示したフローチャートはその構成上第12図のフローチャートと基本的に同じで、第12図のステップS80、82、S84に代つて、ステップS95で「入力」キー81からの座標入力指示の有無を確かめるものである。

#### (カーソル移動速度の可変化)

以上説明した実施例は移動指示領域40又は60に入力したときは原則的に、グラフィックカーソルの移動速度は指の移動速度と同じものであつた。次に説明する実施例はこのカーソルの移動速度を可変にするものである。これは、カーソルを移動するに際し、目標が遠いときはカーソルを高速に移動し、目標点に近づいたときは精度を上げるために指の移動速度より遅くして近づけるものである。

そのために、第10図、第12図、第13図のステップS68及び第11図のステップS118を、第14図の如く変更する。即ち、前記フローチャートのステップS68又はステップS128へきたときは、第14図のステップS1001へジャンプして、実行する。まず、指の移動変位をそれぞれx座標、y座標でx、yとすると、

ステップS1001で、

$\delta x \leftarrow$

(仮座標90のx座標値) - (旧座標93のx座標値)

$\delta y \leftarrow$

(仮座標90のy座標値) - (旧座標93のy座標値)

を算出する。そして、ステップS1002で、グラフィックカーソルの移動量を $\gamma$  (107)として、

$$\gamma \leftarrow \sqrt{(\delta x)^2 + (\delta y)^2}$$

を算出し、メモリ6に格納されていた一定距離の閾値R (111)と比較する。移動量 $\gamma$  (107)がR (111)未満の場合、ステップS1003を実行し、そうでないときステップS1003を省略してステップS1004に進む。

ことにより、たとえ座標入力装置の分解能が小さくても任意の位置への移動が可能となる。一般に正確な位置を指示するときは、初めはすばやく大体の位置合わせをして、最終位置を指示するときはゆつくりと行なうこととする人間の行動とも合致しているため違和感もない。

さて第14図のプログラムは定期的に呼び出される。従つて、上述した式により計算される移動量は、その単位がドットであるにもかかわらず、そのプログラムの呼出しの周期性故、速度としての意味をもつ。ここで、前記一定の閾値R (111)を導いてみる。

カーソルの移動速度が毎秒30mm未満の移動が微調モード、これ以上を粗調モードとする事が操作者に適当な操作感を与えるものであるならば、この30mm内には、画面のドット密度が4ドット

ステップS1003では、移動量 $\delta x$ 、 $\delta y$ にそれぞれ定数 $\alpha$  (108)を乗じた結果をあらためてそれぞれの移動量とする。ここで、定数 $\alpha$  (108)は1未満とすることにより実際の指の動きに比較して移動量を小さく変更することになる。1以上の量であれば逆に高速にする事ができる。ステップS1004にて、グラフィックカーソルの基準座標( $x_0$ ,  $y_0$ )を

$$\begin{cases} x_0 \leftarrow x_0 + \delta x \\ y_0 \leftarrow y_0 + \delta y \end{cases}$$

とする。

以上のように構成すると、正確な位置の指示は移動指示領域に触れて、そのまま指を動かして4つのL字形領域の中心、すなわち代替位置をすばやく目的位置近傍に移動させたのち、正確に位置合わせをするときは、ゆつくりと指を移動させる

／mmとした場合に、4×30ドットが含まれる。即ち、カーソルの移動速度を $v$ とすれば、 $v = 120$ ドット／秒である。従つて、座標入力プログラムが $t$ 、秒毎に呼び出されるとすると、

$$R = t \cdot v = t \cdot (30 \times 4)$$

$$= 120t \text{ [ドット]}$$

が閾値R (111)の目安となる。例えば、座標入力プログラムを100mm毎に呼び出した場合、 $R (111) = 12$  [ドット]となる。このときの指の移動が $\gamma$ ドットであつたとき、指の移動速度 $v$ は

$$v = \gamma / t \text{ [ドット/秒]}$$

である。ここで、 $\gamma$ の単位は[(画面の辺距離) / (その長さの座標間の長さ)]である。

(移動速度可変の変形例)

さて座標入力プログラムを100mm毎に呼び出

した場合閾値  $R = 1.2$  となるのは前述した通りであるが、サンプリングの度（座標入力プログラムが呼び出される度）に指の移動量が  $1.2$  未満のときは速度が  $30 \text{ mm/s}$ 。未満との判断ができる。そこで座標入力の分解能が低い場合の対処を以下に説明する。このような場合でもカーソルの代替位置は  $1$  ドット毎に精度よく特定の点を指示できくてはならない。そこで、 $n$  ドット（ $n$  は自然数）毎の座標入力しかできないようなタッチパネル  $1$  の場合でも（例えば、最小幅が  $1 \text{ mm}$  のとき  $4$  ドット毎の座標入力となる）、上記の定数  $\alpha$ （ $10.8$ ）を「 $1/n$ 」に設定すれば、入力変位が  $\gamma$  のとき、ゆつくり移動したときの代替位置座標の移動は  $\gamma/n$  となり、最小移動  $n$  ドットの場合、代替位置座標の変位は  $1$  ドットとなるので、 $1$  ドット単位に任意の位置を指示することが可能とな

をステップ  $S1002$  で用いられよう。

次に、グラフィックカーソルの操作の1態様として指の移動のし方が急激に大きく移動する場合がある。その対処として、第14図のフローチャートにかえて第15図のフローチャートの制御を用いてもよい。即ち、ステップ  $S1001$  のつぎにステップ  $S1010$  で移動量  $\gamma$  を

$$\gamma = \sqrt{(\delta x)^2 + (\delta y)^2}$$

で算出し、ステップ  $S1011$  で  $\gamma$  が「 $0$ 」かどうかチェックし、「 $0$ 」以外の場合、ステップ  $S1012$  で平均変位  $\gamma'$  を

$$\gamma' = \gamma / \beta$$

で算出する。このパラメータ  $\beta$ （ $109$ ）の初期値は値「 $1$ 」としておく。 $\gamma = 0$  のときは、ステップ  $S1014$  で平均変位  $\gamma'$  も「 $0$ 」にして、ステップ  $S1015$  では  $\beta$ （ $109$ ）の値を  $1$  つ

る。

ただし、上記例の場合は毎秒  $10$  回のサンプリングの例（ $100 \text{ ms}$  のサンプリングレート）を示したが、さらにスムーズな動きを実行する場合や、さらに高速に軌跡を入力したいときなどはサンプリングレートを上げる必要がある。こういった場合、サンプリング毎の変位が小さくなってしまい、上述の方法では変位速度が高速、低速の区別がつかなくなる。例えば、 $10 \text{ m}$ 、毎のサンプリングにすると、 $R = 1.2$  となつてしまい、最小  $4$  ドット毎の座標入力の場合  $\gamma \geq 4$  となるべきであるが、 $\gamma < R$ （ $1.11$ ）の条件は  $\gamma = 0$  のときしか成り立たなくなつてしまい、上記の効果がなくなってしまう。このような場合は1つの方法として、過去  $k$  回のサンプリングにおける変位の平均値を  $\gamma'$  として算出して記憶し、 $\gamma' < R$  の条件

インクリメントしておく。こうすることにより、 $\gamma$ （ $107$ ）が「 $0$ 」から急激に大となつても、 $\beta$ （ $109$ ）は  $\gamma$ （ $107$ ）が「 $0$ 」であつた時間に見合った量となつていたので、ステップ  $S1012$  で  $\gamma$ （ $107$ ）/ $\beta$ （ $109$ ）とする事により急激な移動を緩和することができる。又、ステップ  $S1013$  で  $\beta$ （ $109$ ）を「 $1$ 」に設定し直すことにより、 $\gamma$ （ $107$ ）が連続して大きな値のときは、前記変位の緩和は最初だけであるのでその変位に見合った変位  $\gamma'$  が次回から得られる。上記の手法の概念を第16図に示した。

このようにしておく、座標入力の分解能が低いような場合に、変位「 $0$ 」が続き、ある境界で急激に変位が大きくなる場合にも対応がとれるようになる。

また更に、上記実施例では、変位  $\gamma$  が閾値  $R$ （



111)より小さいとき、水平変位 $\delta x$ 、垂直変位 $\delta y$ に係数 $\alpha$ (108)をかけるようにしたが、係数 $\alpha$ (108)を変位 $r$ の関数 $f(r)$ を定義して、ステップS1002(S1002')を省略し、ステップS1003の代わりに、ここで、

$$\begin{cases} \delta x \leftarrow \delta x \times f(r) \\ \delta y \leftarrow \delta y \times f(r) \end{cases}$$

としてもよい。この関数 $f(r)$ を

$$f(r) = \begin{cases} 1 & : r \geq R \\ 1/n & : r < R \end{cases}$$

として、プログラムを100m。毎に呼び出した場合 $R=12$ となる。よつて、複数個の $\alpha_1, \alpha_2, \dots, \alpha_n$ のうちのいずれか1つを返すようにして、選ばれた $\alpha$ に $r$ を乗してもよい。尚、 $\alpha$ を1未満だけでなく、1以上にすれば、急激に指を動かす、グラフィックカーソルをはじくように動かすことも可能となる。

#### 4. 図面の簡単な説明

第1図は本発明に係る実施例の原理を説明する図、

第2図は従来例の問題点をCADにおいて説明した図、

第3図(a)は実施例の座標入力/表示装置のブロック構成図、

第3図(b)はCRTディスプレイの表示面を説明する図、

第4図は実施例に用いられるグラフィックカーソルの原理的構成図、

第5図(a)～(d)は実施例に用いられるグラフィックカーソルの種々の例の構成図、

第6図(a)～(d)はグラフィックカーソル等を用いて操作を行う様子を説明する図、

第7図(a)は描画プログラムをCADに適用

することも可能となる。

(一般図形への適用)

ところで、以上色々な実施例について説明したものは全てカーソル又はグラフィックカーソルの例であつた。しかし、本発明はカーソル又はグラフィックカーソルへの適用に限まるものではない。即ち、一般的な図形を移動対象とすることも当然考えられる。この場合、上記実施例のカーソル又はグラフィックカーソルを用いて移動対象の図形を指定し、その指定された図形の移動は前述したカーソルの移動と類似した手法により、即ち平行移動により簡単になされる。

[発明の効果]

以上説明したように本発明によれば、移動対象となる図形を所望の目標点に所望の移動速度で正確に移動可能な表示装置を提供できる。

した様子を説明する図、

第7図(b)、(c)は座標入力プログラムと描画プログラムとの関連を説明する図、

第8図は実施例の制御に用いられる種々の制御情報がメモリに格納されている様子を説明する図、

第9図は描画プログラムの一例のフローチャート、

第10図～第15図は実施例の制御手順のフローチャート、

第16図はカーソルの移動の変形例の動作を説明する図である。

図中、

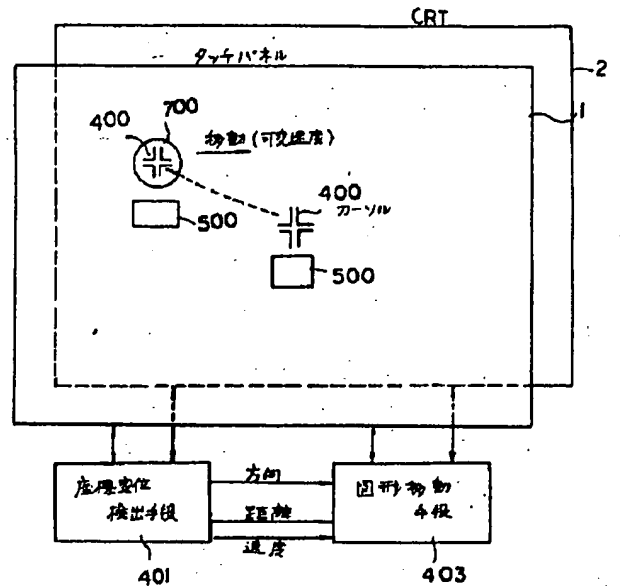
1…タッチパネル、2…CRTディスプレイ、4…CPU、5…画像メモリ、6…メモリ、7…タッチパネル制御回路、9…キーボード、10…

カーソル、11…カーソル移動指示表示領域、12…座標入力指示表示領域、13…座標入力指示入力領域、14…カーソル移動指示入力領域、30…座標入力指示領域、40…移動指示領域、50…座標入力指示領域、60…移動指示領域、100、200、300、600…グラフィックカーソル、C、D…カーソル移動軌跡、P、Q、R、S…代替位置、400…カーソル、401…座標変位検出手段、402…カーソル移動手段、403…図形移動手段、500…カーソル移動指示領域、700…移動対象図形である。

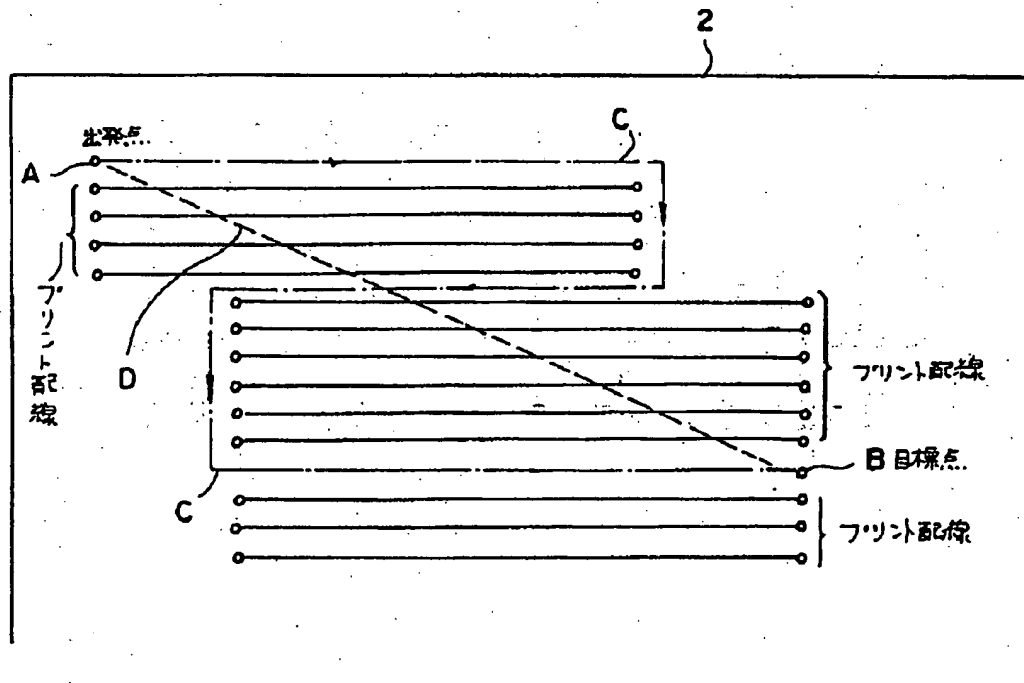
特許出願人 キヤノン株式会社  
代理人 弁理士 大塚康徳



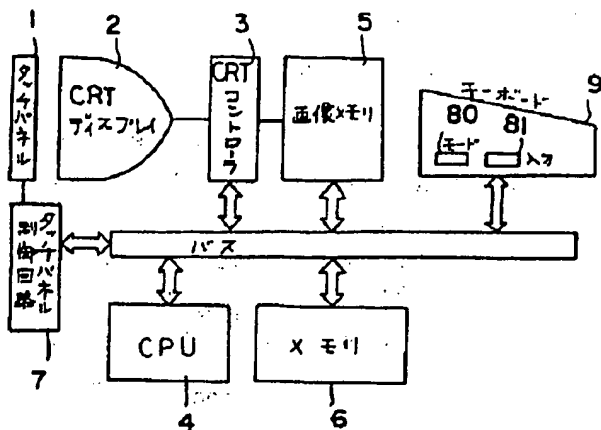
第1図



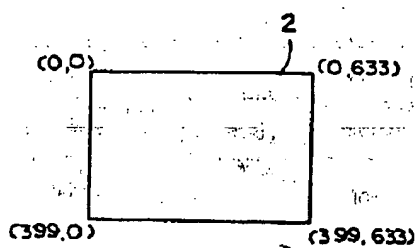
第2図



第3図 (a)



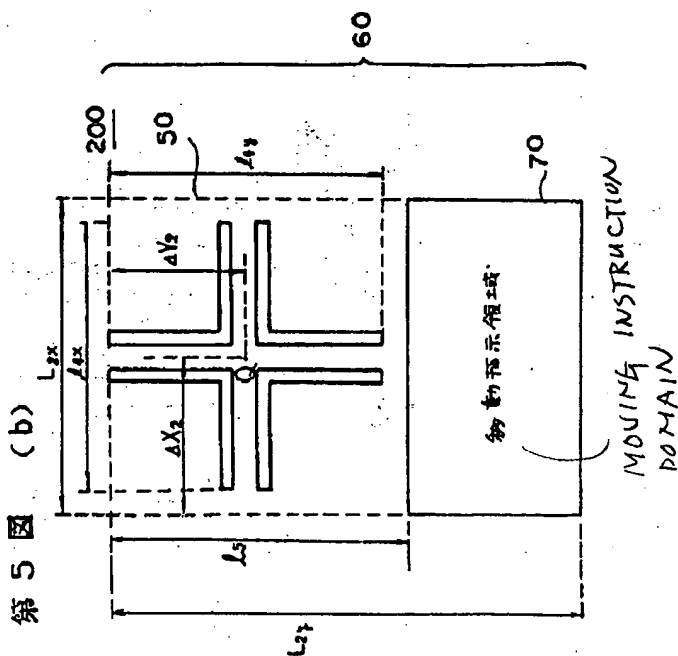
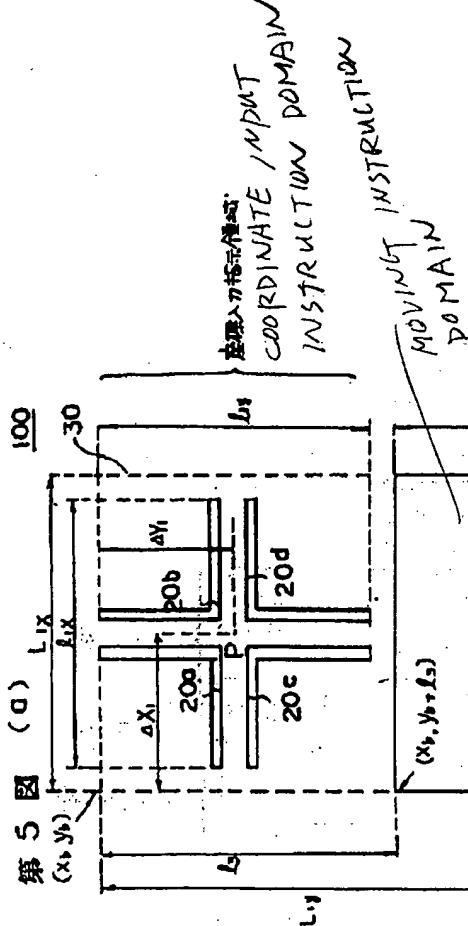
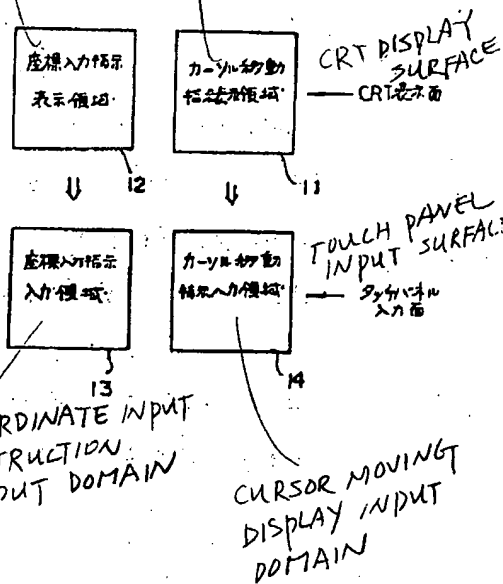
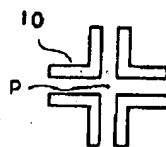
第3図 (b)



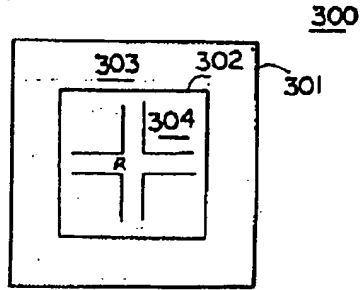
COORDINATE INPUT INSTRUCTION DISPLAY DOMAIN

CURTAIN MOVING INSTRUCTION DISPLAY DOMAIN

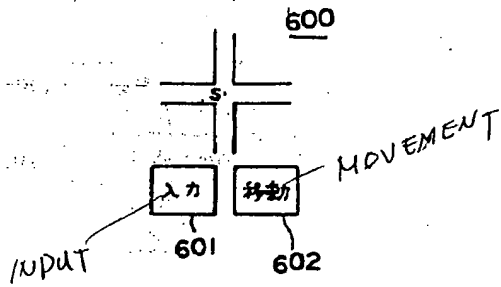
第4図



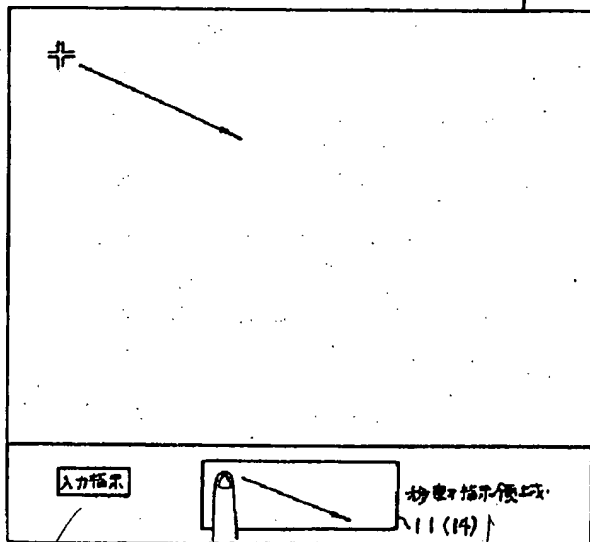
第 5 図 (c)



第 5 図 (d)



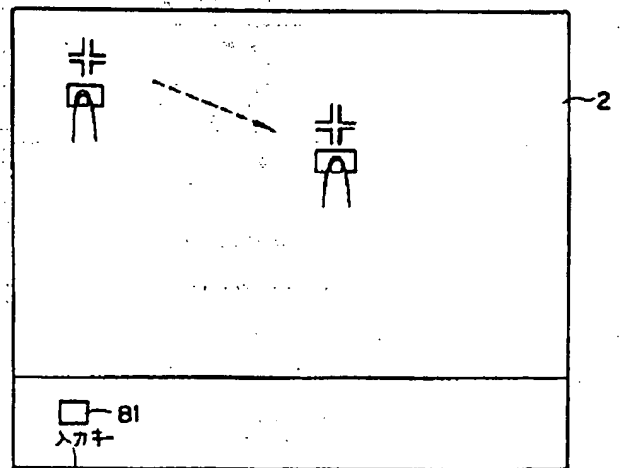
第 6 図 (c)



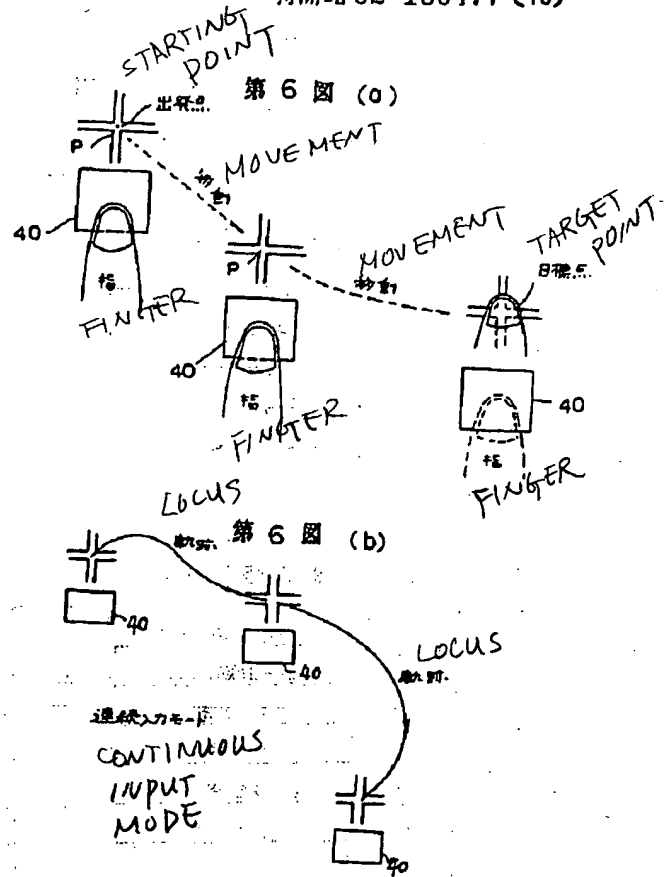
INPUT  
INSTRUCTION

MOVING  
INSTRUCTION  
DOMAIN

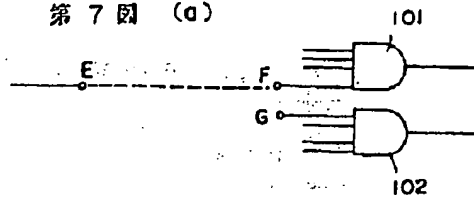
第 6 図 (d)



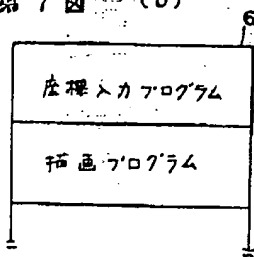
INPUT  
KEY



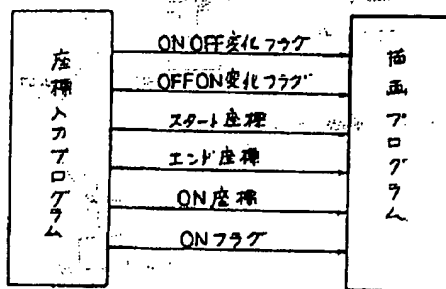
第 7 図 (a)



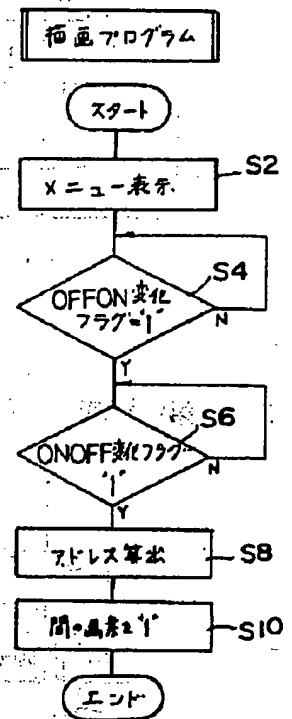
第 7 図 (b)



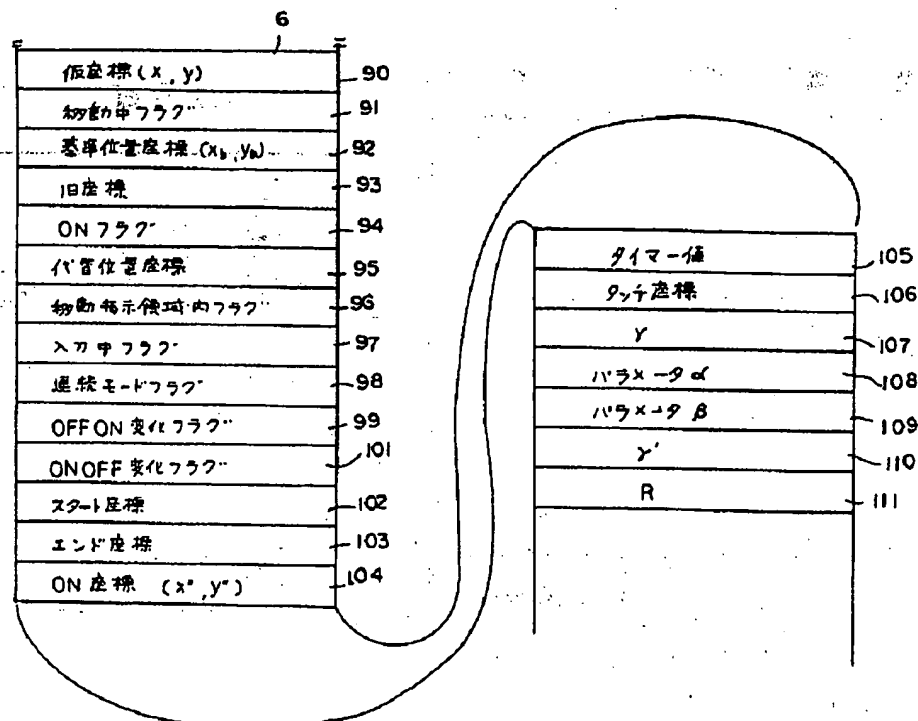
第 7 図 (c)



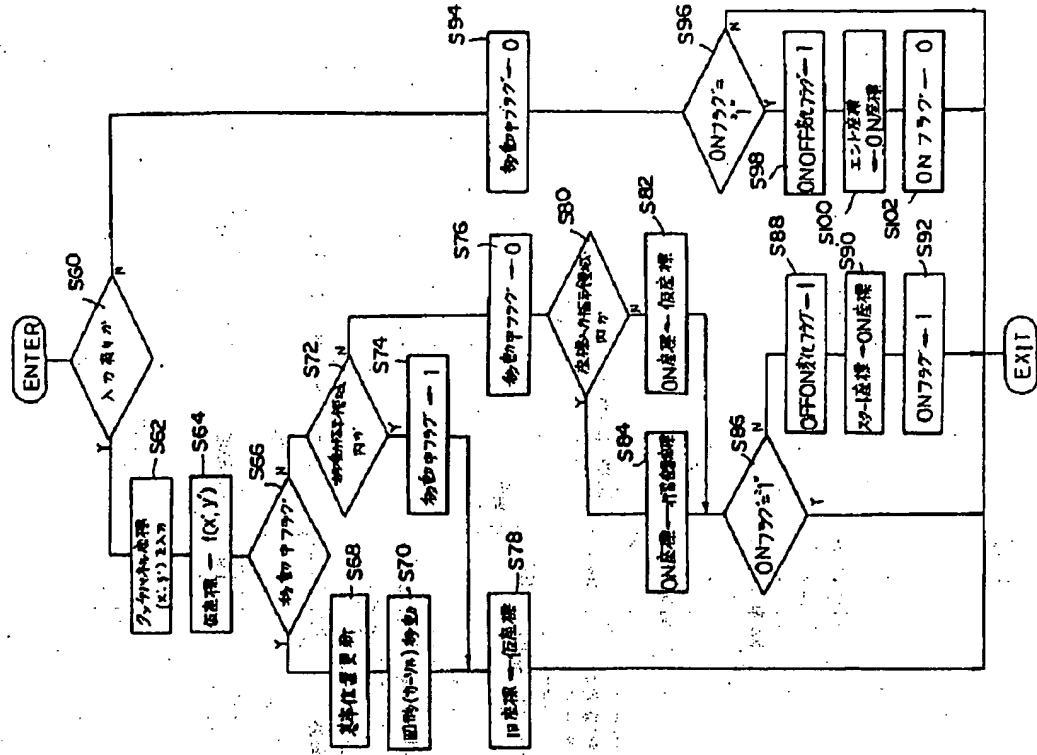
第 9 図



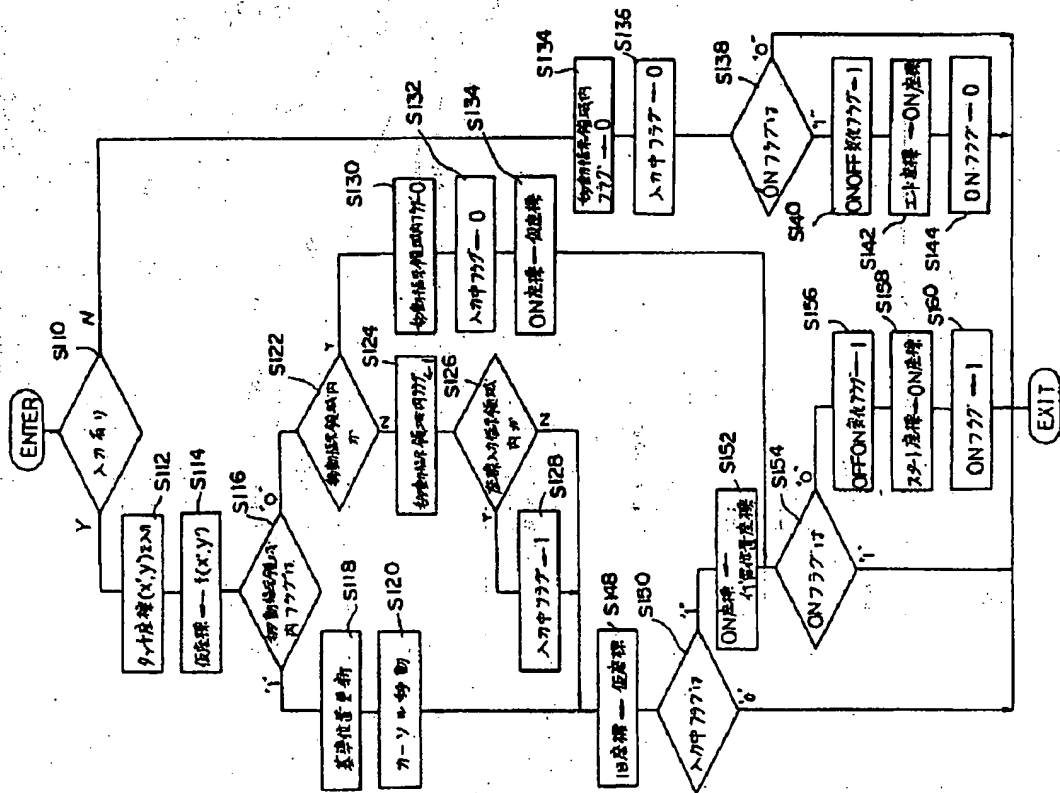
第 8 図



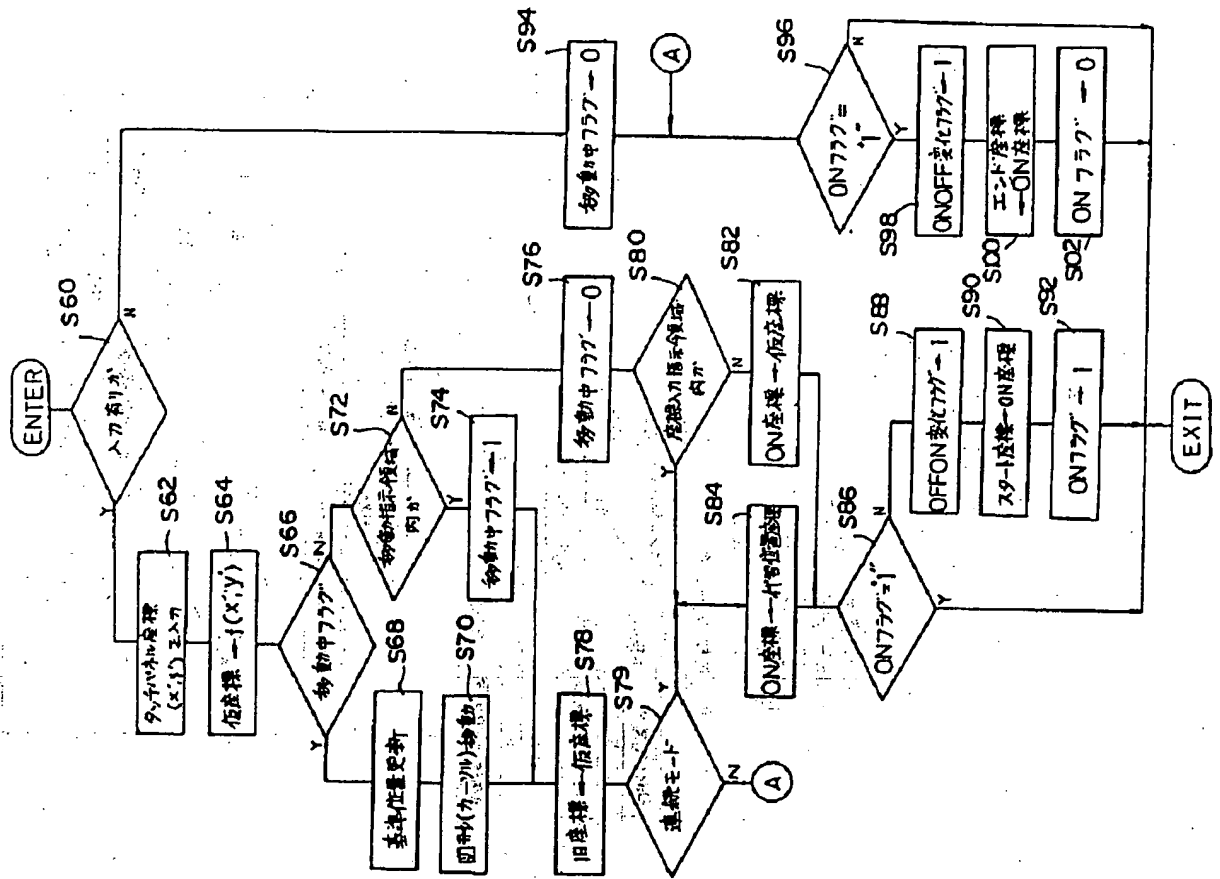
第10図



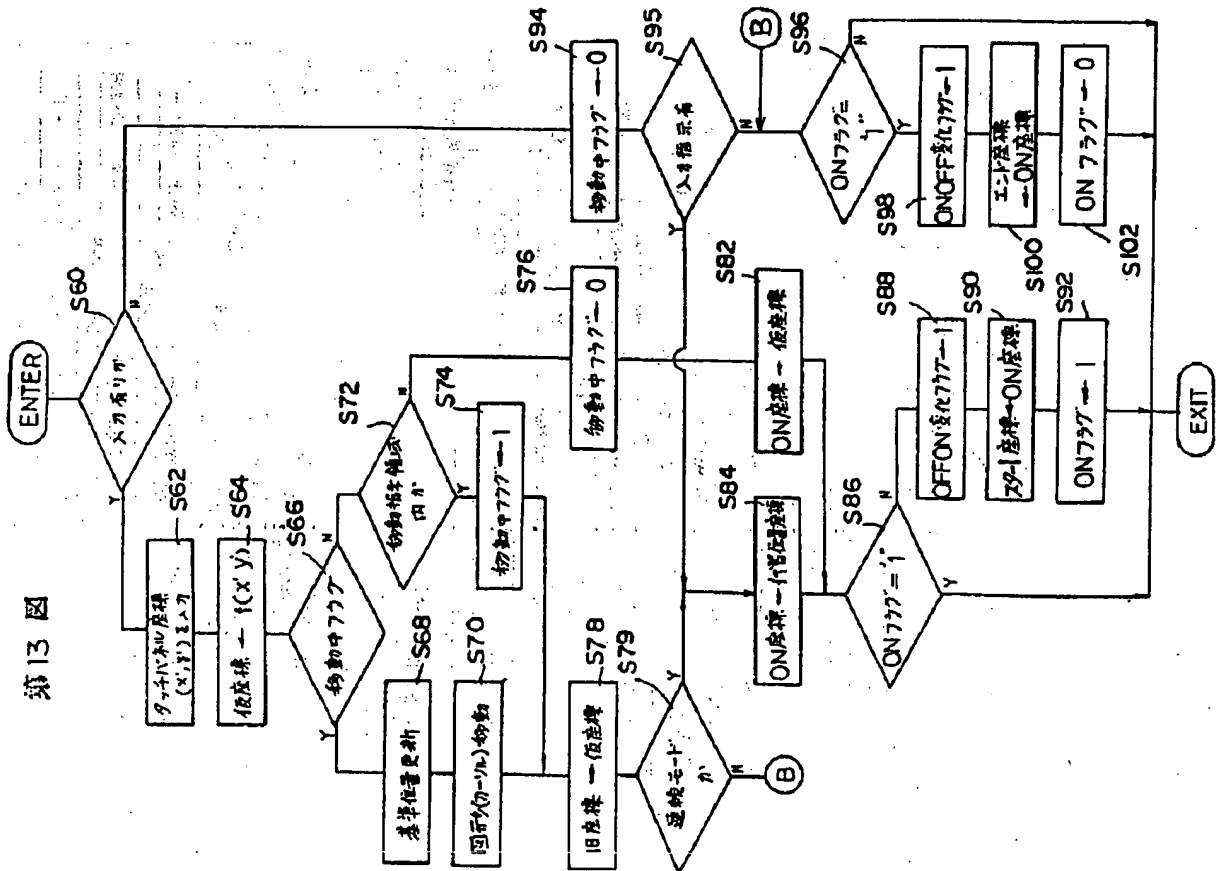
第11図



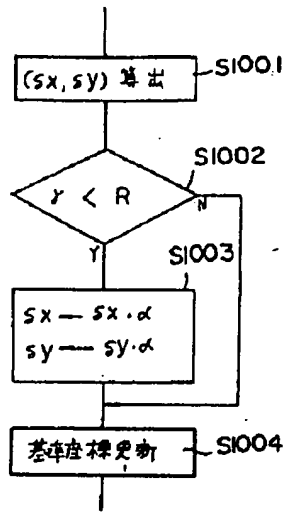
第12図



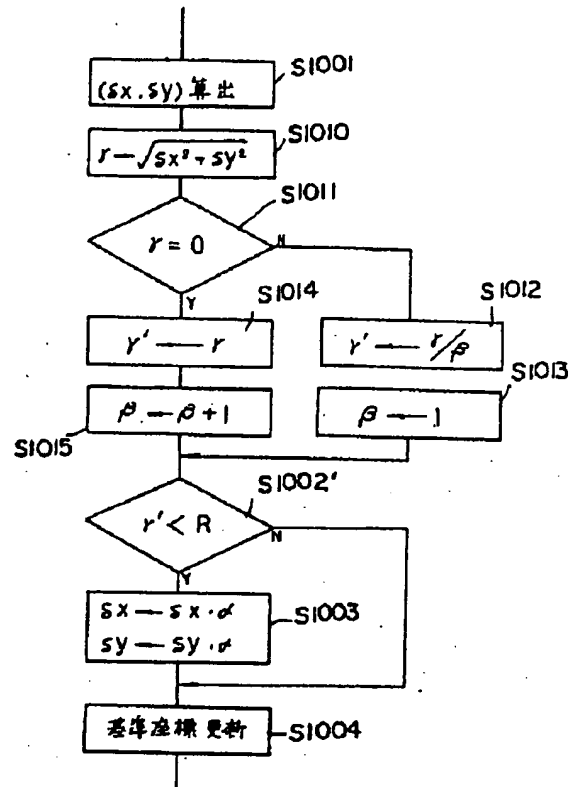
第13図



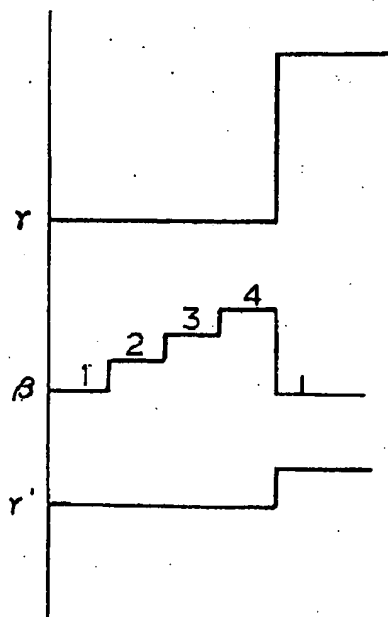
第 14 圖



第 15 圖



第 16 圖





(Graphic Cursor)

Fig. 4 illustrates an example of the principle structure of a graphic cursor used in the preferred embodiment of the present invention. On the display area of the CRT display device 2, the three display domains of a cursor 10 combining L type figures, a cursor moving command display domain 11 and a coordinate input command display domain 12 are provided. Moreover, two domains of the cursor moving command input domain 14 corresponding to the cursor moving command display domain 11 and the coordinate input command input domain 13 corresponding to the coordinate input command display domain 12 are provided at the coordinate input area of the touch panel 1. An operator gives moving direction and moving distance within the cursor moving command input domain 14 while observing the cursor moving command display domain 11, for example, with a finger or stylus pen or the like in order to move the center P of the cursor 10 to the desired display area. Moreover, an operator instructs to input the pixel of the center P of the cursor 10 as the new pixel (dot) by depressing the coordinate input command input domain 13 while observing the display of the coordinate input command display domain 12.

The cursor structured as explained above is called the "graphic cursor" in this embodiment. As illustrated in Fig. 4, the graphic cursor consisting of five structural elements

is combined appropriately to realize various manipulations.  
For example,

① The cursor 10, the cursor moving command display domain 11 and the cursor moving command input domain 14 are set in the approximated domain where the cursor 10 is never hidden even when a finger depresses the cursor moving command input domain 14. When the cursor 10 is moved by moving a finger on the cursor moving command display domain 11, the cursor moving command display domain 11 and the cursor moving command input domain 14 also move together with the cursor 10. In this manipulation, the cursor 10 moves keeping short distance for the finger as the finger moves and moreover the cursor 10 is never hidden with the finger because it is sufficiently separated from the finger. Therefore, in this case, manipulation property is extremely high. Moreover, since the cursor can be moved up to the target position accurately with high efficiency, it assures higher value as a display device.

② The coordinate input command display domain 12 and coordinate input command input domain 13 are also set approximately to the cursor 10, in addition to the structure of ① and when the cursor 10 moves, this domain also moves. In this method, in addition to the characteristic described in ①, manipulation property is extremely improved from viewpoint of the coordinate input (refer to Fig. 5(a) and 5(b) as the example).

③ Above two structures explained above realizes the

graphic display with the cursor moving command display domain 11 and coordinate input command display domain 12. As the alternative display, it is also possible to use, for example, the "input" key 81 or the like on the keyboard 9 in place of the coordinate input command display domain 12 and coordinate input command input domain 13. Moreover,

④ The cursor moving command display domain 11 and cursor moving command input domain 14 are never moved with the cursor 10 but are set in the fixed domain as illustrated in Fig. 6(c).

⑤ Moreover, it is also possible to use an ordinary figure as the moving object in place of the cursor.

(Practical Example of Graphic Cursor)

Two examples of the structure of graphic cursor are illustrated in Fig. 5(a) and Fig. 5(b).

In the graphic cursor 100 of Fig. 5(a), the cursor 10, coordinate input command display domain 12 and coordinate input command input domain 13 are stacked with alignment (such stacked domain is called the coordinate input command domain 30) and the moving command domain 40 obtained by stacking the cursor moving command display domain 11 and cursor moving command input domain 14 is set in the vicinity of the coordinate input command domain 30. The entire part of the graphic cursor 100 of Fig. 5(a) is formed in the rectangular domain having the horizontal length  $L_{1x}$  and vertical length  $L_{1y}$ . The L-shape domains 20a, 20b, 20c, 20d form the cursor 10 and this structure makes easier the identification of the

center part P thereof. In view of assuring distinctive difference between the graphic cursor in the related art and the graphic cursor of the present embodiment, the center of L-shape part at the four sides is called hereinafter the "substituting position". In the characteristic part of the present invention, the cursor 10 is separated from the moving command domain 40 and the coordinate of the center part P of cursor becomes the substituting input coordinate for the input with a finger.

When the reference coordinate position of the graphic cursor 100 is assumed to be located at the upper left position  $(x_b, y_b)$ , the horizontal distance is  $\Delta X_1$  and vertical distance is  $\Delta Y_1$  as the relative length of the substituting position P. Namely, the substituting position P is expressed as  $(\Delta X_1, \Delta Y_1)$  with the position  $(x_b, y_b)$  considered as the origin. The moving command domain 40 is expressed as a square shape and a size of this domain is expressed as the horizontal length of  $l_{2x}$ , and vertical length of  $l_{2y}$  and moreover the left upper position as the reference point of the relevant domain can be expressed as  $(x_b, y_b + l_3)$ . Moreover, the reference position of the coordinate input command domain 30 is defined as  $(x_b, y_b)$  and its horizontal length is determined as  $l_{1x}$  and the vertical length as  $l_{1y}$ . As explained above, it is very easy to draw the desired graphic cursor figure by giving the reference coordinate of each figure and the difference from such coordinate. Namely, when the reference position is moved, the graphic cursor 100 also moves in direct.

Fig. 5(b) illustrates an example of the graphic cursor 200 having the function which is a little different from the graphic cursor 100 of Fig. 5(a). This graphic cursor 200 is formed as a rectangular domain including the horizontal length of  $L_{2y}$  and vertical length of  $L_{2y}$ . Offset of the substituting position Q is  $\Delta X_1$  in its horizontal direction and  $\Delta Y_1$  in its vertical direction. Particular difference from the graphic cursor 100 is that the moving command domain 60 is set for entire part of the graphic cursor 200. Therefore, the coordinate input command domain 50 (this domain is simultaneously the moving command domain) is the remaining domain of the graphic cursor 200 except the square 70 indicating the domain to instruct only the movement from the moving command domain 60. Here, the horizontal length is  $l_{4x}$  and vertical length  $l_{4y}$  in the coordinate input command domain 50 and the offset from the reference position is 0 both in the horizontal and vertical directions. When the coordinate input command domain 50 is moved while it is depressed not to hide the substituting position Q so that this graphic cursor 200 is capable of continuously inputting the coordinate in addition to the function of the graphic cursor 100, the moving command and coordinate input command can be realized simultaneously.

(Manipulation of Graphic Cursor)

Fig. 6(a) illustrates an example of the manipulating method of the graphic cursor 100 (Fig. 5(a)). While the moving command domain 40 is depressed at the cursor starting

position on the CRT display device 2, the finger is moved without release from the display area. With this traveling of finger, the graphic cursor 100 or 200 as a whole moves. This traveling is realized by changing the values of reference coordinate  $(x_b, y_b)$ . Here, it should be noted that the substituting position P is never hidden with the finger during the movement as can be understood from the figure. When the substituting position P reaches the target position, the target position coordinate is input by depressing the coordinate input command domain 30 with a finger.

Fig. 6(b) illustrates use of this device in the continuous input mode (control procedure of Fig. 11) with the manipulation of the "mode" key 80. In this continuous input mode, when the graphic cursor 100 is moved while the finger touches on the moving command domain 40 of the graphic cursor 100, the graphic cursor 100 naturally moves with movement of the finger and simultaneously the coordinate of the substituting position P is also input and thereby the continuous coordinate input can be obtained (locus can be obtained by displaying such coordinate input in direct).

Fig. 6(c) illustrates an example of the structure in which each structural elements of Fig. 4 are individually illustrated. In this case, the cursor 10 is moving on the CRT display device 2 like the related art. Moreover, the cursor moving command display domain 11, coordinate input command display domain 12, coordinate input command input domain 13 and cursor moving command input domain 14 do not.

move and remain in the fixed position of the figure.

Fig. 6(d) illustrates an example where coordinate input command is made with the coordinate input key 81 on the keyboard 9.

(Example of the other graphic cursors)

Figs. 5(c), 5(d) illustrate examples of the other graphic cursors 300 and 600. As illustrated in Fig. 5(c), the domain between the external frame 301 and internal frame 302 is defined as the command domain only for the movement and the internal side of the internal frame 302 is defined as the input command domain 304, while the internal side of the external frame 301 as the moving command domain 303. The substituting coordinate R is also expressed with the center of four L-shape figure as explained above. The substituting coordinate R can be moved freely to the desired position at the upper, lower, right and left end of the display area by designating the input command domain around the substituting coordinate and then designating the moving command domain around such input command domain. As illustrated in Fig. 5(c), it is possible to define the input command domain 601 to the area other than the peripheral area of the substituting target S. Particularly, when the size of graphic cursor is limited, it is effective when locus can be input even from the area isolated from the substituting coordinate S. Of course, it is possible that only the domain 602 is moved, while the domain 601 and the area near the L-shape area are defined as the input command domain.

This is to request you to file an IDS again for U.S. Patent application Serial No.: 09/149,216 (divisional application). Most of the enclosed documents have already been sent to you with our letters of January 8, 1998 and June 17, 1997, requesting to file an IDS for the corresponding parent application. Enclosed are:

(a) An English-language translation of the Japanese Office Action;

(b) Japanese Laid-Open Patent Application No Hei 01-314321, an entire English-language translation of it, an English-language translation of its drawings and an English-language Abstract of its Patent; and

(c) Japanese Laid-Open Patent Application No Hei 03-222033 and corresponding US Patent 5,347,628.

(1) Please file the document of item (a) as the concise explanation.

The part (page 5, lower right to page 6, upper left) of Japanese reference 1 (Japanese Patent Laid-Open No Hei 03-22033) specified in the Office Action as particularly relevant corresponds to column 4, lines 21-33 of US Patent 5,347,628.

(2) Please file the documents of item (b) again. In the corresponding parent application, we were not sure to file; however, file the English-language translation of the drawings now. At this time, specify the publication date clearly as December 19, 1989, because the date was noted just as 12/1989 in the former FORM PTO-1449 LIST.

(3) Please file the documents of item (c) again. At this time, specify the publication date as October 1, 1991, which was noted just as 10/1991 in the former LIST, because it may be especially important to clarify that Japanese Patent Laid-Open No Hei 03-222033 was published before the priority date (October 7, 1991) of the present application.



# WRITTEN NOTICE FOR REJECTION REASON

Patent Application No.: HEI 3-258232  
Date of Origination: May 2, 1997  
Examiner: Shinichi Yamazaki (9174 5E00)  
Agent: Sadakazu Igeta  
Applied Articles: Item 2 of Article 29, Article 36

This patent application should be rejected depending on the reasons explained below. Those who have some claims are requested to present the written argument within 60 days from the mailing date of this notice.

## REASONS

1. The invention in relation to the claims listed below of this patent application cannot be patented depending on the regulations of the item 23 of the Article 29 of the Patent Law because this invention has been attained easily by those skilled in the ordinary technical field of the invention before presentation of the patent application on the basis of the inventions described in the following publication distributed in Japan and/or foreign countries before the application.

NOTE (refer to a list of references for the cited references)

- Claims: 1, 2, 8, 20
- Cited references, etc.: 1, 2
- Remark: , >

Particularly refer to Fig. 4 and its explanation (lower right column on page 5 to upper left column on page 6)

Use of the well known touch panel (reference 2) as the input apparatus is easy for those skilled in this art.

- Claims: 13, 25, 26
- Cited references, etc.: 2
- Remark:

The cited reference 2 describes that the special display may be realized by touching on the object through the touch panel.

The specification and drawings of this patent application do not satisfy the conditions specified in item 4 or 5 and 6 of the article 36 in the following points.

#### NOTE

- Claim: 6
- Remark:

The other claim is not uniquely cited and unclear.

- Claim: 16
- Remark:

It is unclear which claim is cited. (Meaning of "said object" and "said scroll speed" is unclear.)

For the inventions in relation to the claims other than those suggested in this Written Notice for Rejection Reasons, any reason for rejection is not found at present. If some reasons for rejection are newly found, the reasons for rejection will be noticed.

#### List of Cited References

1. Official gazette of Japanese Patent Application Laid-open  
No. HEI 3-222.033

2. Official gazette of Japanese Patent Application Laid-open  
No. HEI 1-314321